

Books

Communications for Cooperating Systems: OSI, SNA, and TCP/IP, R. J. Cypser, Addison-Wesley Publishing Company, Inc., Reading, MA (1991). 768 pp. (ISBN 0-201-50775-7).

Editor's note: The following two reviews of this book are from quite different perspectives. The first is by an expert in the field; the second is by an experienced technologist who is learning about computer networks and communications. Cypser's book is the latest addition to the Systems Programming Series, published by Addison-Wesley in cooperation with IBM.

Seldom in my years as a networking systems professional have I seen as broad and ambitious an undertaking as *Communications for Cooperating Systems: OSI, SNA, and TCP/IP*—100 000 pages of architectural specifications from the combined milieus of OSI, Internet, and SNA condensed into less than 800 pages. This monograph has a bit of something for everyone concerned with multiprotocol support in the IBM systems environment. For the applications technologist are discussions of applications interfaces and cooperative processing models; for the network technologist are discussions of network elements and technologies from the LAN/WAN link layer to the bridge/routing layer and above; for the administrative technologist are discussions of network and systems management.

Coming from a strong single-vendor, single-protocol background (Cypser authored the original tome on SNA, *Communications Architecture for Distributed Systems*, Addison-Wesley, 1978), it may be surprising that the fundamental hypothesis for this book is the requirement for protocol and system diversity. In observing the diverse

evolutionary nature of real cooperative systems, the author notes "the result is not homogeneity ... it is a mixture of proprietary advances and ... standards." In the treatment which ensues, it is refreshing to note both a bent toward architectural pragmatism (concurrent OSI, SNA, and TCP) and a hope of (appropriate) architectural convergence. While the author does not achieve the panacea of universal convergence of architectures (Who can?), he does manage to bring together in a structured fashion much material that has previously been scattered.

This book is not meant to be read end-to-end, in one sitting, but rather topically, in bits and pieces. Some of my favorite sections cover common programming interfaces (CPI-CC, CPI-RR) and layer-by-layer naming services (X.500 directory to APPN-n network layer to LAN discovery protocols). I also like the little tidbits and insights into IBM techno-futures, such as the hints of CMIP running over SNA transport and SNA addressing evolving to global OSI addressing.

The book does have a few weak points. While OSI and SNA are balanced throughout, the treatment of TCP is not. The collection of all SNA and OSI architectures into a meta-architecture, OSNA, also is perhaps a bit too ambitious. But these weaknesses are compensated by other strengths. The strongest element in the book, in fact, may be the discussion of the various bridging, routing, "brouting," and "grouting" taxonomies in the section on transport service providers. In fact, turn to this section first, and read forward and back from there.

In 768 pages, only so much can be covered. Don't expect a detailed bottom up treatment of all ar-

©Copyright 1991 by International Business Machines Corporation.

chitecture families and cooperative processing paradigms. Don't expect critical tradeoff analyses comparing overhead and throughput of OSI, SNA, TCP, and NetBios. These analyses are not the focus of the book. The focus is rather the correlation of architectures as peers, and the convergence of architecture in hierarchies.

The book does leave the reader with a sense that despite the complexity of a mixed-vendor, mixed-architecture protocol environment, rational convergences with SNA are indeed possible. Whether via hierarchical encapsulation among dissimilar protocol families, converged access to common subnet layers, or access to common application programming interfaces, a few rays of hope yet glimmer that multiple protocol networks, while incredibly complex, are somehow manageable. (But we purists know that a single protocol architecture is still best!)

The author has done a credible job of treating a very complex subject. For professionals interested in the technical issues and trends concerning the pragmatic convergence of multiple protocol architectures, especially the context of SNA, this book is recommended reading.

John R. Pickens
3 Com Corporation
Santa Clara
California

Like many information systems professionals, I have developed a technical expertise in large systems and in storage systems to address the requirements of the last 25 years. Changes in recent years make it necessary to expand that technical base into the communications arena. Both changing customer requirements and IBM's refocused efforts to provide products and a structure to meet those requirements make it necessary to understand cooperation and communication capabilities in a multivendor world. The challenge to assimilate technical information in this new environment is shared by many other I/S professionals.

The necessity to learn quickly, but with the opportunity to depend upon specialists for technical support, make it advantageous to avoid a series of product-oriented, technical education classes. *Communications for Cooperating Systems: OSI, SNA, and TCP/IP* meets my need to understand the world of communications in today's environment. The author has researched the many facets of this complex world and has provided a single source for a very concise, technical, and organized body of information.

Chapters 1 and 2 form a general understanding of the enterprise requirement for communication in a multivendor environment, including dependence upon the ability to exchange information. Current structures that meet those enterprise needs are presented, as well as additional requirements. The discussion solidifies an understanding of both today's requirements and the future benefits of a solution designed to provide any-to-any communication. Open Systems Network Architecture (OSNA) is defined as an evolving macroarchitecture of heterogeneous systems, embracing multiple communication protocols and providing the communications foundation of SAA[™] and AIX[®]/OSF systems, and also facilitating communication with still other systems. The requirement for use of standards, while taking advantage of technical flexibility, is discussed.

Architectural structures, discussed in Chapter 3, provide a definition of the layered approach in information processing, with a focus on the OSI structure. The concepts of logical connections and use of formats and protocols provide the backbone for later chapters.

Chapters 4 through 6 define some of the application services that support cooperating systems. The presentation of the issues and challenges associated with the management of multiple subnetworks, different architectures, and multiple vendors affirms the need for a coordinated management approach. The overview of the resource managers that assist in a distributed systems operation provides additional insight into the relationships between the different "partners" in a total solution. The capabilities designed for the interchange of text, voice, and messages re-enforce the requirement for implementation of standards across all platforms. Of particular interest is the discussion of distributed data systems. The func-

tions defined provide an exciting set of opportunities for solving today's problems.

Chapters 7 through 11 define a lower-level set of interfaces that provide application programmers with facilities that support the end-to-end exchange of data. The discussions do not define completeness of capabilities but inform the reader about the potential for creating capabilities with a set of building blocks generally available. The expanded view of transactions available with LU 6.2, presented in Chapter 10, provides an exciting transition to the distributed world.

The next four chapters following Chapter 11 contain detailed information concerning the transport service providers in the lower layers of the architecture. OSI, TCP, NetBios, SNA/Subarea, and SNA/APPN are presented. Of significant benefit are the discussions that define what is necessary for end systems to connect to multiple types of transport service providers, as well as for multiprotocol service providers (e.g., a backbone that can carry X.25, SNA, TCP/IP, and NetBios traffic). The discussion of the ease-of-use of APPN provides an exciting description of the capabilities of a self-defining network.

The lower layers are presented in Chapters 16 through 18 with a discussion of the architectures, specifics of local area networks, and high-speed technologies supporting today's performance requirements.

The last chapter is a reflection of where we are today—and where we are going. The examples of systems and subsystems working together are impressive. The reader can certainly envision capabilities that, a few years ago, would have been seen as impossible.

After reading *Communications for Cooperating Systems: OSI, SNA, and TCP/IP*, am I now an expert? No! But I have become very aware of issues and requirements and of the solutions available today. The book brings together an immense body of information. The first reading provides background, awareness, and the ability to explore. The second (and *n*th) reading will make use of the book as a reference document, providing pointers to required, specific information.

Cypser has been my researcher. He has put together information pertinent to today's world of communication. Thanks!

Nancy L. Ellis
IBM US Marketing & Services
St. Louis
Missouri

SAA is a trademark, and AIX is a registered trademark, of International Business Machines Corporation.

The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill, John M. Carroll, The MIT Press, Cambridge, MA (1990). 340 pp. (ISBN 0-262-03163-9).

The Nurnberg Funnel is a thorough and carefully crafted tale of the pursuit of effective self-teaching techniques for inexperienced users of computer systems. It is the story of the failure of traditional educational techniques, predicated on filling up the mind with knowledge (hence the Nurnberg funnel), to adequately educate novice computer users. In addition it recounts the evolution of a program of research from empirical evaluation of computer usage to the proactive, psychologically motivated interface design of new computer artifacts. Finally, it depicts the integration of quantitative empirical research with sensitivity to the *in situ* experience of the user.

Contrary to expectations and popular belief, even graphical user interfaces are baffling to newcomers, and contrary to fairly pervasive assumptions about education in general, and computer manuals, tutorials, and help in particular, comprehensive explanations and detailed, step-by-step instructions are insufficient to remedy the situation. In fact, they may make it worse.

What Carroll observes is that adults use computer systems for a reason: to write a letter or memo, for example. They know much about performing these real world tasks, and want to get as directly as possible to their real work. He finds that they reason about the computer system in terms of

what they already know (which doesn't always work), that they are inclined to try to *do* things (ready or not), that they read only sporadically (often not in the order anticipated by the author), and that often they get lost in recesses of the system (the books, believing users will do what they are told, don't anticipate errors).

Carroll's objectives are to (1) get learners started immediately on meaningful tasks, (2) minimize reading and other passive training, and (3) turn errors into opportunities for learning. The book describes the experiments and projects that uncovered the need for a minimalist and task-oriented approach to software education, and that evolved and exemplify such an approach in the design of software and instructional systems.

This is a book to come back to repeatedly: when embarking on design, when building instructional materials, when performing usability tests, when summarizing results for publication. There is so much here that should be remembered—so much that is still not part of our collective understanding of what it means to interact with a computer. Carroll describes hazards novices encounter working within the desktop metaphor that may be as relevant today as they were during his work with the Lisa. His experiences with baffled users resonate disconcertingly with user testing of more "advanced" systems, both on a large scale, and even more distressingly, in detail.

Linda Tetzlaff
IBM Research Division
Hawthorne
New York

An Introduction to Database Systems, Vol. 1, fifth edition, C. J. Date, Addison-Wesley Publishing Company, Inc., Reading, MA (1990). 854 pp. (ISBN 0-201-51381-1).

It is interesting to note that the first edition of this book, which appeared in 1976, measured six-tenths of an inch thick. The current edition, recently published, has grown to one and four-tenths inches thick.

This increase in size reflects the increase in the breadth and depth of material and topics that currently fall under the heading of "database systems," especially, of course, relational database.

The author, C. J. Date, is widely known as one of the earliest proponents and expositors of the relational approach (as originated by E. F. Codd, IBM Fellow, retired), as well as a heavy contributor to the continuing evolution of both the theory and practice of relational database. Here he continues commendably to perform the service of keeping his regular readers current, as well as introducing the novice to the field. It is enough to say that with this edition of a book widely used in both university and industrial education programs, the author adheres to his usual high standards for clear exposition. For this reviewer, the book also provides the student with an accurate view of where we are today with database technology and a good indication of where the field is headed in the near future.

As is the case in previous editions of this book, Date takes on the difficult job of explaining very abstract and comparatively complex topics. Again, he performs this task with admirable clarity and precision without sacrificing readability. Much of the material is either new or enhanced—so that this really is a *new* edition of this introduction. Those familiar with earlier editions can still benefit from a careful reading of this new work.

Among the topics that are entirely new for this edition are object oriented and logic based database systems. Those in need of tutorials on these areas of somewhat recent interest in database management will be rewarded well by time spent on these chapters.

The book provides a brief introduction to what has become the *de facto* (as well as ANSI designated) standard relational database language, SQL, along with overviews of two relational database systems: IBM's DATABASE 2™ (DB2™) and INGRES™ (recently acquired by ASK, Inc.). There are appendices on IMS, DATACOM/DB, and IDMS as examples of hierarchic, inverted list, and network DBMSs respectively.

The book also constitutes a good, brief introduction to distributed database, explaining well the basic problems associated with concurrency con-

trol for such systems. Last, but certainly not the least merit of the book, is its comprehensive, lucid, and persuasive treatment of the key characteristics of the relational database approach and of relational database theory. Beginners interested in what appears to be the database approach of choice for the 1990s will profit much from this recent addition to the Addison-Wesley Systems Programming Series.

The book contains useful and instructive exercises as well as very valuable annotated bibliographies.

Charles J. Bontempo
IBM Thornwood Education Center
Thornwood
New York

DATABASE 2 and DB2 are trademarks of International Business Machines Corporation.

INGRES is a trademark of ASK, Inc., formerly held by Ingres Corporation.

Note—The books reviewed are those the Editor thinks might be of interest to our readers. The reviews express the opinions of the reviewers.